

\*\*\*\*\*

ECET 3810

Laboratory Exercise 4

\*\*\*\*\*

Objective: Introduce the student to the basic design and implementation of interfaces and associated classes. Specifically, the student will design a class to implement an existing interface and modify a separate class to consume the services offered by the interface.

Procedures:

i) Start TextPad

ii) An interface is like a billboard that specifies certain services offered by an implementing class. For instance: if a class called Washer had 10 different methods, but you were only interested in advertising 3 of them to another consumer class, then you would create an interface containing only the methods to be advertised. Copy the following Java source code into the code editor pane:

```
public interface IPower  
{  
    public double getPower();  
    public void setPower(double d);  
}
```

iii) Notice that the methods (`getPower` and `setPower`) within the `IPower` interface have no body; that is they end in a semicolon and have no curly braces. Since they have no body, they are referred to as abstract. Save the copied code to a file named `IPower.java` Note - in the “Save As” dialog box the “Save as type:” dropdown box should be “Java (\*.java)”

iv) Under the “Tools” menu select “Compile Java.” If your code doesn’t compile properly, your tools (SDK and TextPad) may not be configured properly, or something has been misspelled, deleted, or accidentally inserted. Carefully compare the code in your code editor with the code on this page.

v) Copy the following Java source code into the code editor pane:

```
public class Washer implements IPower
{
    // declare our global variables. Note: these are referred to as
    // instance variables, because each instance of this class has
    // its own copy of these variables in memory.
    private double power = 0;
    public Washer(double power)
    {
        this.power = power;
    }
    public double getPower()
    {
        return power;
    }
    public void setPower(double d)
    {
        power = d;
    }
}
```

vi) Save the copied code to a file named Washer.java Note - in the “Save As” dialog box the “Save as type:” dropdown box should be “Java.” Compile as above.

vii) Copy the following Java source code into the code editor pane:

```
import java.util.Scanner;
public class ApplianceApp
{
    private static Scanner sc = new Scanner(System.in);
    public static void main(String[] args)
    {
        System.out.println("Home Appliance Application");
        System.out.print("Enter Power Setting: ");
        IPower washItf = (IPower)new
Washer(sc.nextDouble());
        System.out.println("\nWashing Machine Power is " +
washItf.getPower() + " watts");
    }
}
```

viii) Save the copied code to a file named ApplianceApp.java Note - in the "Save As" dialog box the "Save as type:" dropdown box should be "Java." Compile as above.

ix) With ApplianceApp.java selected in the Document Selector pane click the "Tools" menu and select "Run Java Application." Verify that the application runs correctly.

x) Once you have verified that the program is working properly, create a new class called Dryer.java, which also implements IPower. Add two new abstract methods to the IPower interface called getName and setName. Hint: by adding these two new abstract methods to the interface what other file(s) must be modified?

xi) Modify ApplianceApp by adding a user prompt for the dryer's power setting, creating the specified dryer object, and printing out the dryer's power by using an interface object to call the dryer's getPower methods.

xii) Add code to ApplianceApp using your washer and dryer interface objects to set the name of each appliance. Then add additional code to print out the appliance names.

xiii) Properly annotate all of your files, including the addition of prologs.

**Turn in:** A cover sheet (course, lab number(s), your name, due date), introduction (at least two paragraphs introducing the lab and your modifications), screen captures of your functioning code, a copy of your modified source code, and a conclusion (at least two paragraphs stating such things as: main skill(s) you learned, troubles you encountered, etc.). Do not turn in unstapled labs. Plan ahead and find a stapler.